



MSA UNIVERSITY
FACULTY OF COMPUTER SCIENCE
Cloud Computing Course Project
VirtCloud



A Cloud-Based Virtual Machine Management Platform

User Manual

Supervised by:
Dr. Mohamed Hassan Mahmoud Elgazzar

Team Members:

- Mostafa Hesham - 220367
- Mahmoud Ibrahim Elmolla - 224417
- Youssef Ahmed Hany - 220427
- Mahmoud Ahmed - 237957
- Youssef Abdelhamid - 223339
- Karim Emad Helmy Shaker - 220995

Contents

1	Introduction	2
2	Getting Started	2
2.1	Sign Up	2
2.2	Log In	2
3	Plans and Credits	3
4	Dashboard Overview	4
5	Creating a Virtual Machine	4
5.1	Step 1: Create Disk	4
5.2	Step 2: Configure Resources	5
5.3	Step 3: Review and Launch	6
6	VM Management	6
7	Disk Operations	7
7.1	View Disk Info	7
7.2	Resize Disk	7
7.3	Convert Disk Format	8
8	VM Runtime and Cost Tracking	8
9	Running the VM	8
10	Docker Management	9
10.1	Navigating to Docker Dashboard	9
10.2	Show Guide (First-Time Users)	9
10.3	Create Dockerfile	10
10.4	Edit Dockerfile	11
10.5	Build Docker Image	11
10.6	View Docker Images	12
10.7	Search Docker Images	12
10.8	Run Docker Container	13
10.9	Manage Running Containers	13
10.10	View Running App	14

1 Introduction

VirtCloud is a user-friendly cloud virtualization platform designed for creating, managing, and operating virtual machines (VMs) through an intuitive web interface. Users can create virtual disks, allocate computing resources, launch operating systems, and track usage, all backed by a credit-based billing system.

2 Getting Started

2.1 Sign Up

To begin using VirtCloud, users must first create an account.

- Navigate to the **Sign Up** page.
- Enter a unique **username**, a valid **email address**, and a **password** (minimum 8 characters).
- Upon submission, your account will be created and assigned a default **Free Plan** with 15 monthly credits.

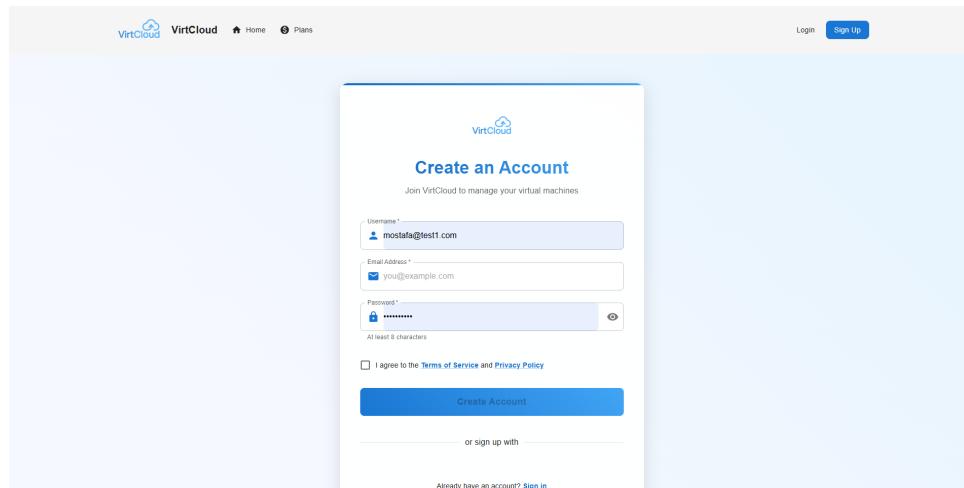


Figure 1: Sign Up Page

2.2 Log In

If you already have an account:

- Navigate to the **Log In** page.
- Enter your **email** and **password**.
- After successful login, you will be redirected to the **Dashboard**.

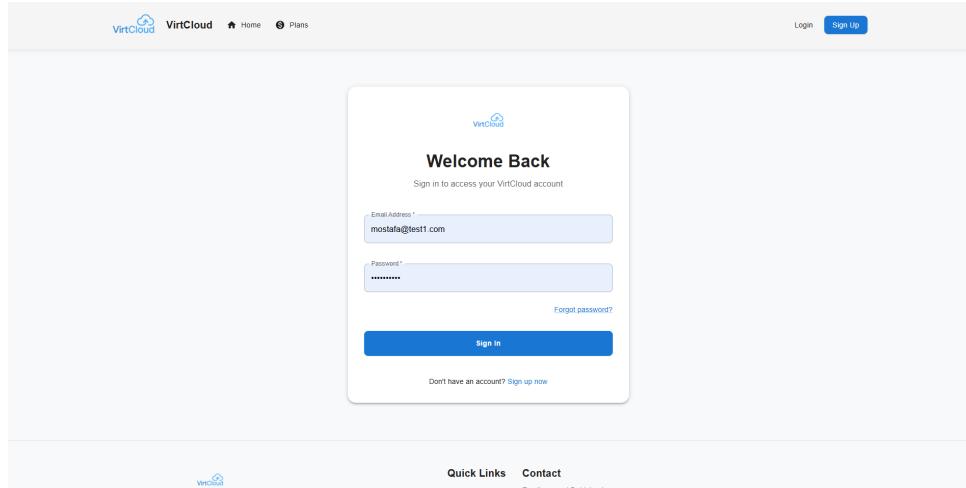


Figure 2: Login Page

3 Plans and Credits

VirtCloud offers four main plans:

- **Free Plan:** 15 credits/month. Up to 2 CPUs, 2GB RAM, 20GB disk, max VM runtime 4 hours.
- **Pro Plan (\$9/month):** 150 credits/month. Up to 4 CPUs, 8GB RAM, 50GB disk, unlimited VM time.
- **Unlimited Plan (\$29/month):** 600 credits/month. Up to 8 CPUs, 16GB RAM, 200GB disk, persistent VMs.
- **Pay-as-you-Go:** Rechargeable plan (\$5 = 10 credits). No limits on usage. Flexible billing per hour based on resource consumption.

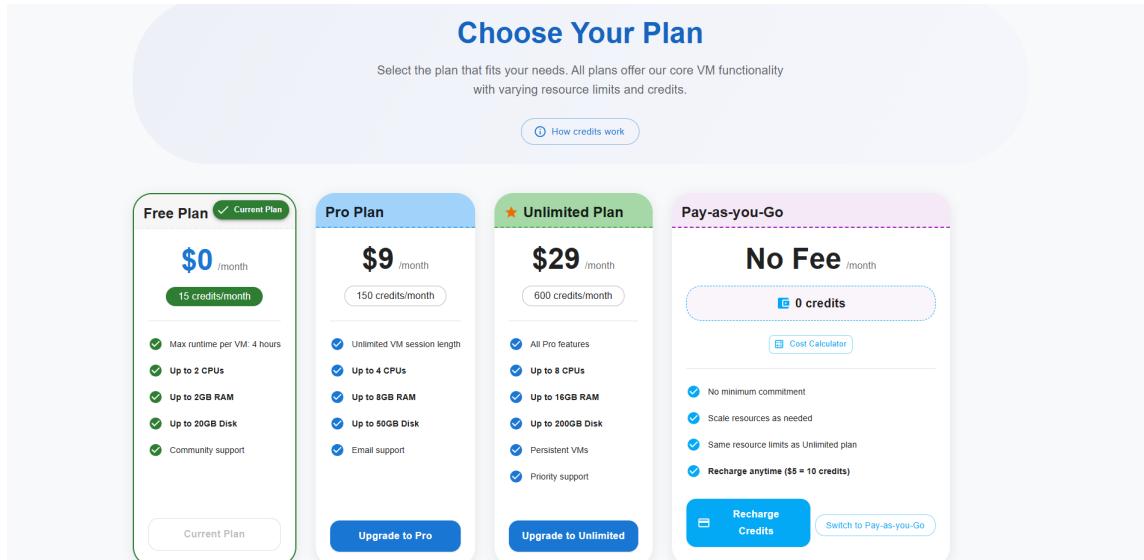


Figure 3: Plan Selection Page

4 Dashboard Overview

The dashboard gives a complete view of your current VMs, available credits, usage, and quick access buttons.

- View total and active VMs.
- Check remaining credits and usage cost.
- Access **Create VM**, **Manage VMs**, and **Refresh Data**.

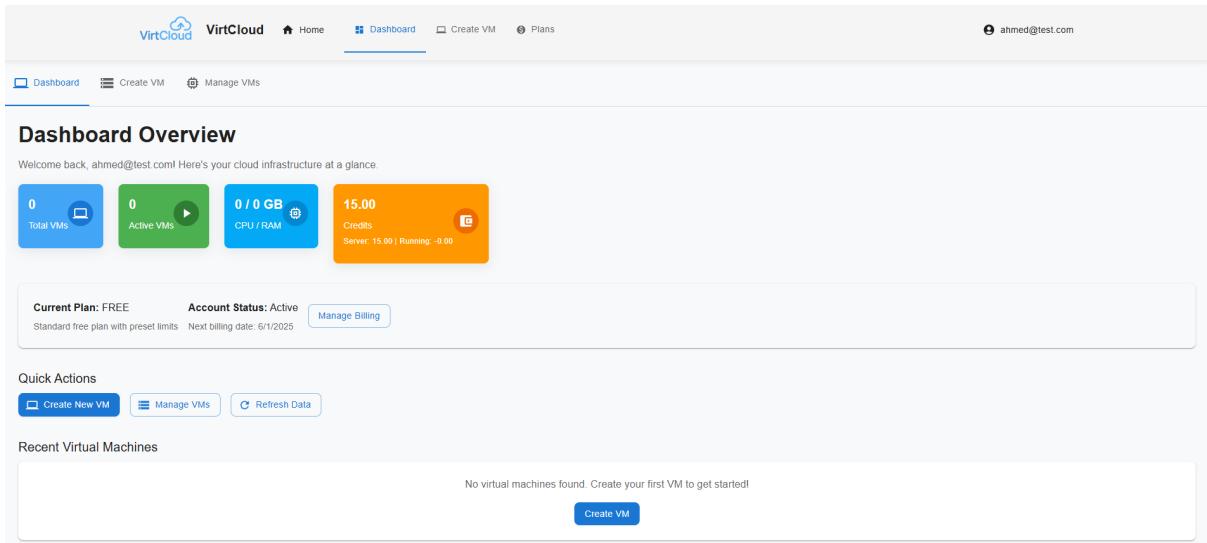


Figure 4: User Dashboard

5 Creating a Virtual Machine

5.1 Step 1: Create Disk

- Enter the **Disk Name** (no extension needed).
- Choose a disk format (e.g., `qcow2`, `raw`).
- Select disk size within your plan's limits.

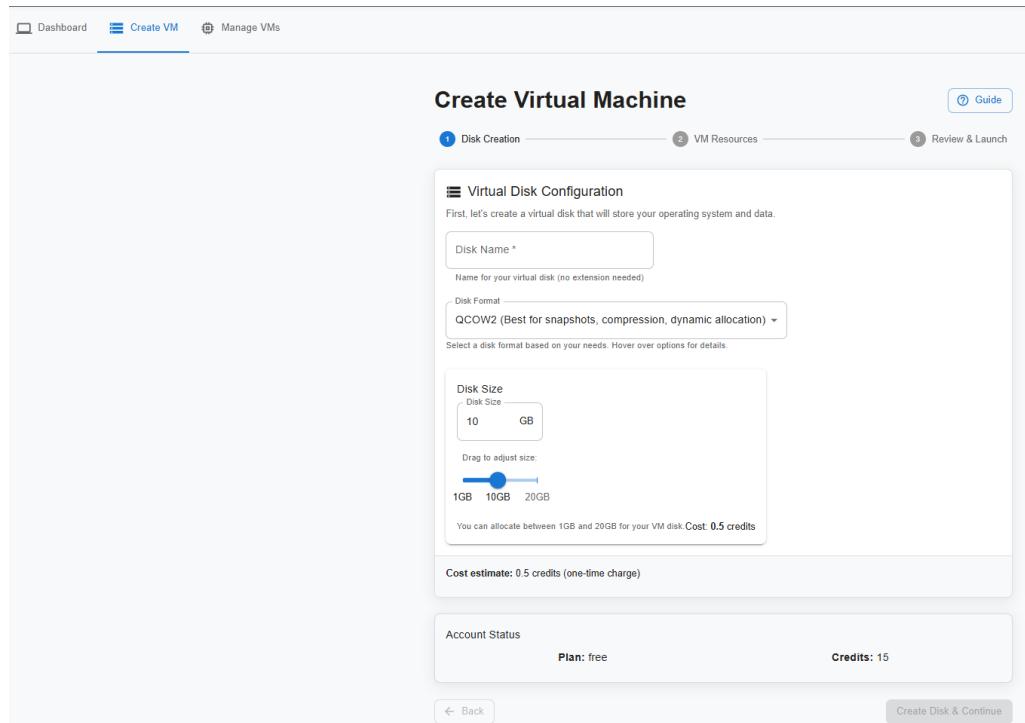


Figure 5: Disk Creation Step

5.2 Step 2: Configure Resources

- Allocate **CPU cores**, **RAM size**, and select **Display type**.
- Select the **Operating System ISO** to install.

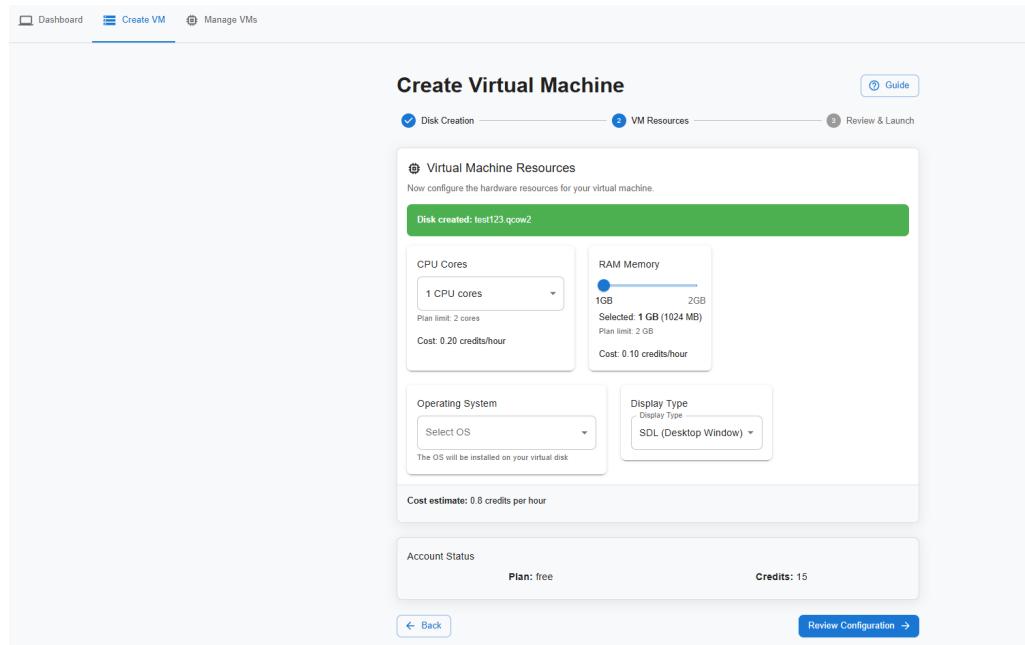


Figure 6: VM Resource Configuration

5.3 Step 3: Review and Launch

Review all settings before launching the VM. Summary includes disk cost and hourly VM usage cost.

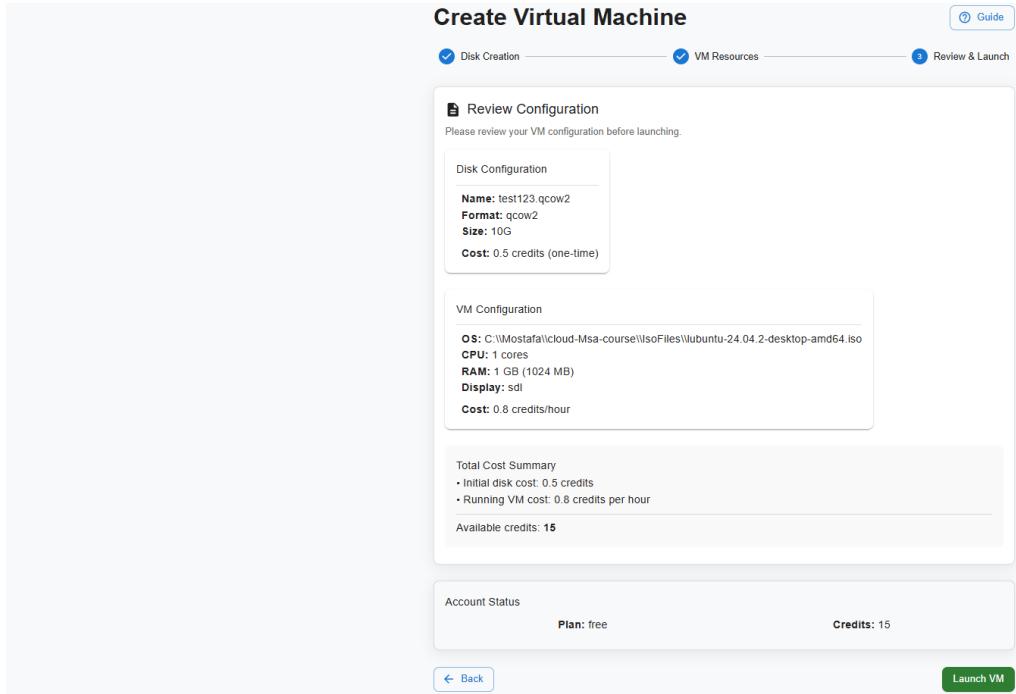


Figure 7: VM Configuration Review

6 VM Management

From the **Manage VMs** page, users can monitor, update, and control their virtual machines.

- **Start/Stop VMs.**
- View runtime, cost, and resources.
- Perform actions like **Disk Info**, **Resize**, **Convert**, and **Rename**.

VM Management

Manage your virtual machines and disk operations.

Current Credits: 15.00 (Server: 15.00) **Total Running Cost:** 0.01 credits/hour

VM Name	Disk	CPU	RAM	Created	Status	Actions	Elapsed	Cost
test123	test123.qcow2	1	1 GB	5/2/2025	running		00:00:40	0.01

Disk Operations Guide

- [Disk Information](#)
- [Resize Disk](#)
- [Convert Disk Format](#)

View detailed information about your disk including format, size, and virtual size. Increase your disk size to accommodate more data. Specify the amount to add. Convert between disk formats (qcow2, raw) while preserving all data.

Example: +5G adds 5 gigabytes to current size

Figure 8: VM Management Interface

7 Disk Operations

7.1 View Disk Info

Check format, actual size, virtual size, and other metadata.

Disk Information: test123.qcow2

General Information

File Location: Unknown Format: Encrypted: No Backing File: None

Size Information

Virtual Size - 10 GiB Actual Disk Size - 193 K Cluster Size - 65536 bytes

Disk Space Usage - 0% Qcow2 format uses space efficiently, allocating only what is needed

Advanced: Raw Command Output

Figure 9: Disk Information Output

7.2 Resize Disk

Extend an existing disk to a larger size. You must specify how much to increase.

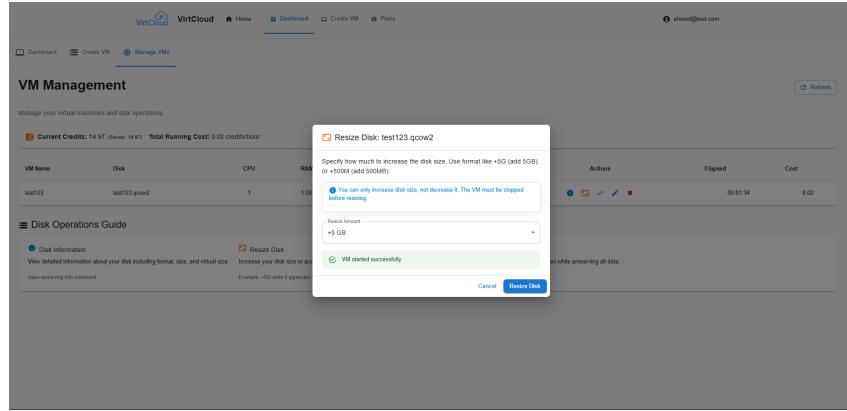


Figure 10: Resize Disk Panel

7.3 Convert Disk Format

Convert from one format (e.g., `qcow2`) to another (e.g., `raw`).

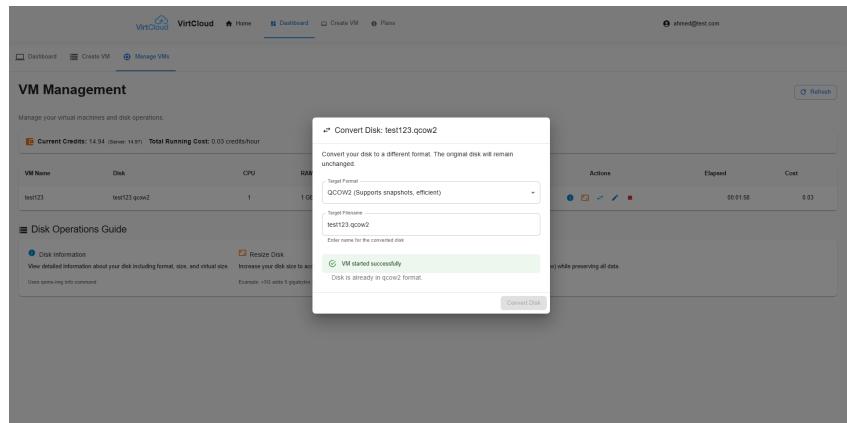


Figure 11: Convert Disk Format Page

8 VM Runtime and Cost Tracking

View credit usage and total runtime cost per VM.

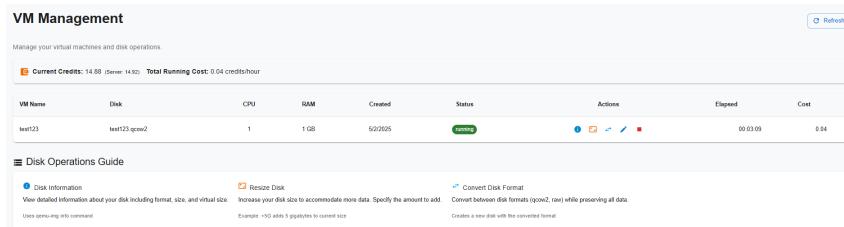


Figure 12: Credit Monitoring and Runtime Cost

9 Running the VM

VMs are launched in a new window. Example below shows Lubuntu booted successfully.

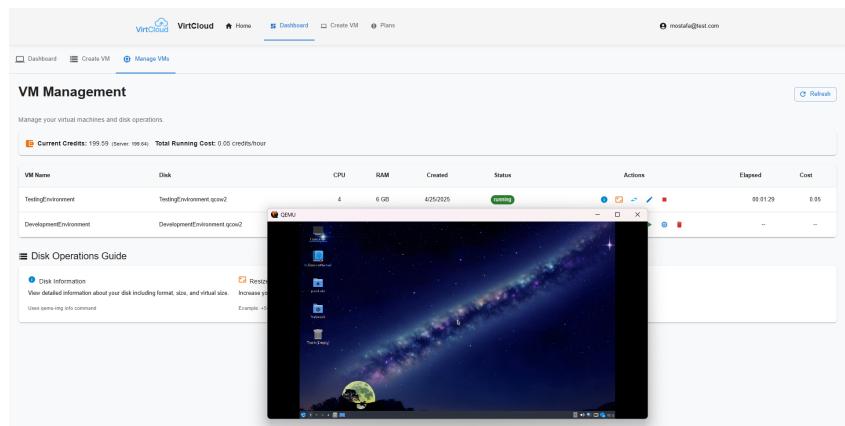


Figure 13: Running Lubuntu Inside VirtCloud

10 Docker Management

VirtCloud now supports Docker-based container management. Users can create Dockerfiles, build images, pull from DockerHub, and run containers — all within their personal cloud environment.

10.1 Navigating to Docker Dashboard

To access Docker features, click the **Docker Dashboard** from the main navigation bar. As shown below, the Docker tab is highlighted in green.

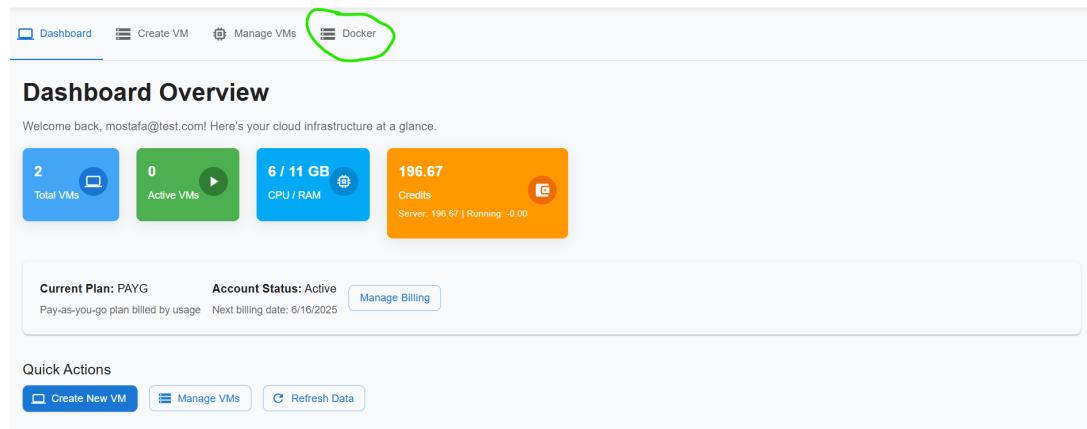


Figure 14: Accessing Docker Dashboard

10.2 Show Guide (First-Time Users)

First-time users can click the **Show Guide** button to learn about available Docker functionalities.

The screenshot shows the VirtCloud Docker Management interface. At the top, there's a navigation bar with links for Home, Dashboard, Create VM, Plans, and a user account. Below the navigation is a secondary navigation bar with links for Dashboard, Create VM, Manage VMs, and Docker. The main content area is titled "Docker Management". On the left, there's a sidebar titled "Dockerfile Management" with a brief description and a table of available operations:

Operation	Description	Icon
Create Dockerfile	Create a new Dockerfile with custom content	+
View Dockerfile	Examine the content of an existing Dockerfile	info
Build Image	Build a Docker image from a Dockerfile	magnifying glass
Delete Dockerfile	Remove a Dockerfile from your system	trash

On the right, there are three buttons: "Docker 28.1.1" (green checkmark), "Refresh" (blue), and "Hide Guide" (with a green circle around it). Below the sidebar, there's a "Docker Management Guide" section with a brief introduction and a table of available operations.

Figure 15: Show Guide for New Users

10.3 Create Dockerfile

To create a Dockerfile:

- Enter a Dockerfile name (e.g., PythonWebapp).
- Add a short description (e.g., flask).
- Provide Dockerfile content.

The screenshot shows the VirtCloud Docker Management interface with the "Create Dockerfile" dialog open. The dialog has fields for "Name" (PythonWebapp) and "Description (Optional)" (Flask). Below the dialog, the main interface shows a list of existing Dockerfiles: test1 and nginx-custom. The "Dockerfiles" tab is selected in the sidebar. On the right, there are "Actions" buttons for each Dockerfile.

Figure 16: Dockerfile Creation

Sample Dockerfile Content

```
FROM python:3.9-slim
WORKDIR /app

RUN pip install --no-cache-dir flask requests
```

```

COPY . /app
RUN echo "No files to copy, creating app in Dockerfile"

RUN echo 'from flask import Flask, request
import socket, os, datetime

app = Flask(__name__)

@app.route("/")
def home():
    host_name = socket.gethostname()
    host_ip = socket.gethostbyname(host_name)
    return f"<html>...dynamic HTML content...</html>

if __name__ == "__main__":
    app.run(debug=True, host="0.0.0.0", port=80) > app.py

EXPOSE 83
CMD ["python", "app.py"]

```

10.4 Edit Dockerfile

Users can click the **Edit** button (highlighted in green) to update Dockerfile content.

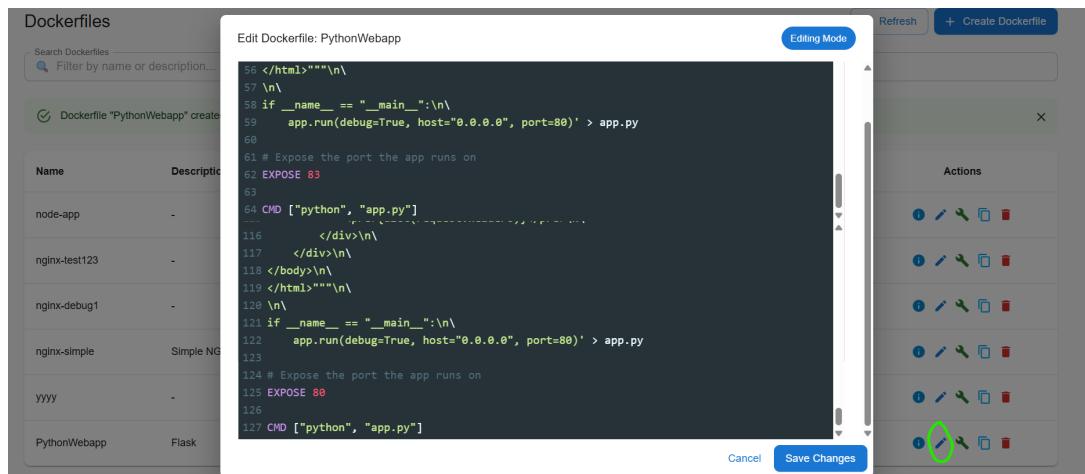


Figure 17: Editing Dockerfile

10.5 Build Docker Image

After preparing a Dockerfile, users can click the **Build Image** button. Provide an image name and tag (e.g., `pythonwebappimage:v1.0`).

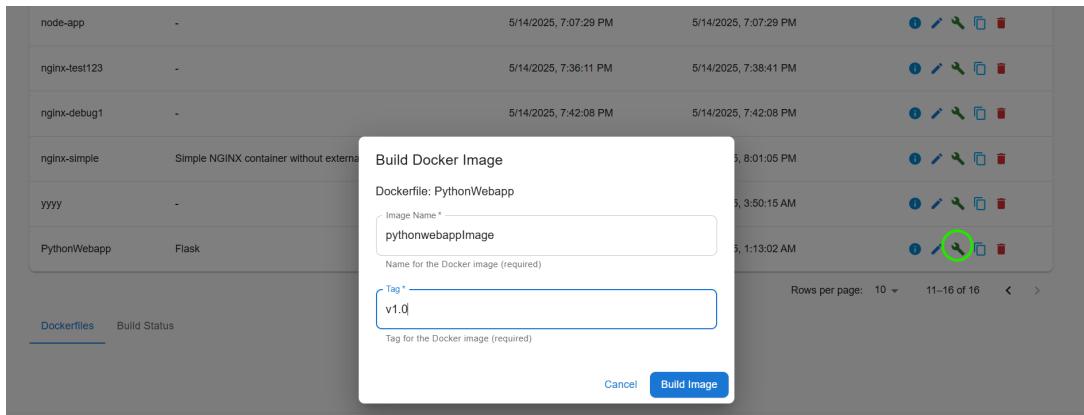


Figure 18: Build Docker Image

10.6 View Docker Images

All created or pulled Docker images are listed under the Images tab.

The screenshot shows the 'Images' tab selected in the top navigation bar. Below it, a search bar and filters for 'Local' and 'Search Local' are visible. A table lists one image entry:

Repository	Tag	Image ID	Size	Created	Ownership	Actions
pythonwebappimage	v1.0	sha256:0d7bf279a15e	49 MB	5/17/2025 4:19:12 AM	My Image	

Figure 19: Docker Image Repository

10.7 Search Docker Images

Users can filter images by:

- **Local Search:** Find locally built images.
- **DockerHub Search:** Pull from public repositories.

The screenshot shows the 'Images' tab selected. In the search bar, the text 'pythonwebappimage' is entered and highlighted with a green box. The 'Local Search Results' tab is selected. The results table shows one entry:

Repository	Tag	Image ID	Size	Created	Actions
pythonwebappimage	v1.0	sha256:0d7bf279a15e	49 MB	5/17/2025 1:26:14 AM	

Figure 20: Searching Local Docker Images

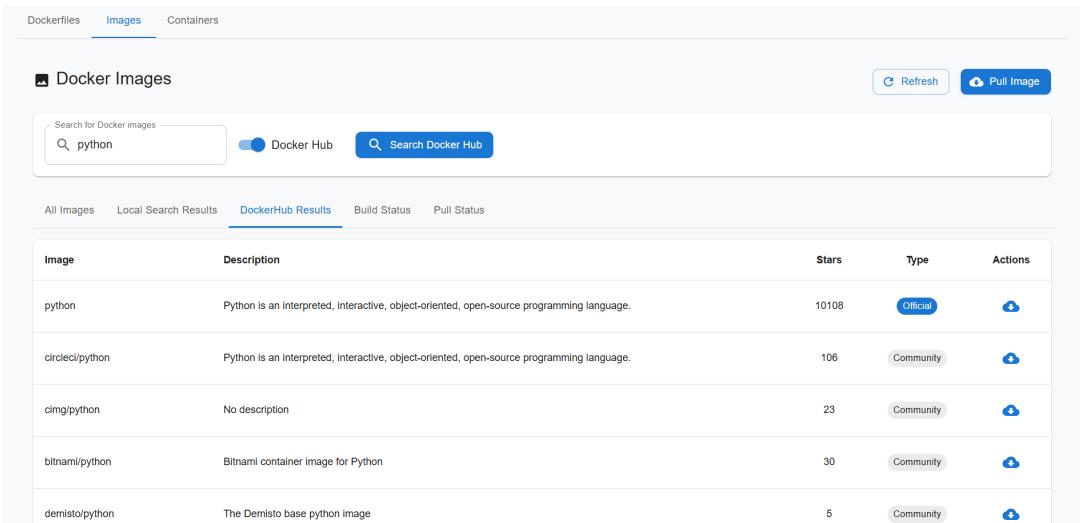


Figure 21: Searching DockerHub and Pull Option

10.8 Run Docker Container

To run a container:

- Click the **Start** button for the desired image.
- Provide a container name and port to expose (e.g., 8000).

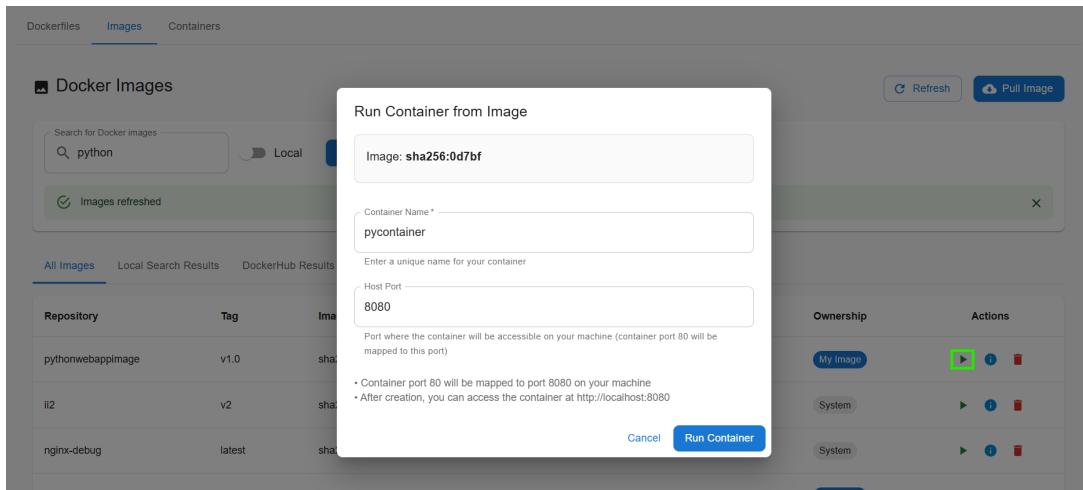


Figure 22: Run Docker Container

10.9 Manage Running Containers

Users can manage running containers from the Containers tab. Actions include stopping and deleting.

The screenshot shows the VirtCloud Docker Management interface. At the top, there are navigation links: Home, Dashboard, Create VM, Plans, and a user account icon. Below the navigation is a secondary navigation bar with links: Dashboard, Create VM, Manage VMs, and Docker. The main title is "Docker Management". A sub-header says "Manage Docker containers, images, and build custom Dockerfiles." Below this are tabs for Dockerfiles, Images, and Containers, with "Containers" being the active tab. The main content area is titled "Docker Containers" and lists two entries:

Name	Image	Created	Status	Ports	Actions
pycontainer	pythonwebappimage:v1.0	17 seconds ago	running	80/tcp → 8080	Stop
laughing_snyder	nginx:latest	6 days ago	exited	None	Start

Figure 23: Running Containers Management

10.10 View Running App

If your container exposes a web app (like Flask), visit the configured port (e.g., `localhost:8000`) to access it.

The screenshot shows a browser window with the URL `localhost:8080`. The page title is "VirtCloud Test Container". A green checkmark icon indicates "Container is running successfully!". Below this, under "Container Information:", the following details are listed:

- Hostname: db77c4c0241e
- IP Address: 172.17.0.2
- Current Time: 2025-05-17 01:38:24
- Python Version: Python 3.9.22

Below this section is another one titled "Request Headers:" which contains the following JSON data:

```
{
  "Host": "localhost:8080",
  "Connection": "keep-alive",
  "Sec-Ch-Ua": "\"Chromium\";v=\"136\", \"Google Chrome\";v=\"136\", \"Not,A/Brand\";v=\"99\"", "Sec-Ch-Ua-Mobile": "?0"
}
```

Figure 24: Container Web Application Output